CLAIMS

1. A method of transforming a class, comprising a usage method accessing at least one class field, said class being loadable by a class loader in an object-oriented environment, said method comprising the steps of:

(a) creating from an original class, which comprises a class field, an original-class class-initialization method, and a helper class, by

i) converting at least one said class field to an instance field and introducing the instance field into said helper class; and

ii) converting the original-class class-initialization method to a helper-class instance-initialization method and introducing it into said helper class which comprises a helper-class class-initialization method; and

(b) creating for the class a corresponding modified class by converting the usage method to a modified-usage method, wherein each access to the class field is replaced by an invocation of an access function for fetching, for a process with an instance of the helper class, from the instance, the instance field corresponding to the class field,

the helper class and the modified class being loadable the class loader.

2. The method according to Claim 1, further comprising the steps of:

(c) creating, for the original class, a corresponding modified-original class; and

(d) converting each original-usage method of the original class to a modified original-usage method and introducing it into said modified-original class, wherein each access to said class field is replaced by an invocation of the access function for fetching, for a process with an instance of said helper class, from the instance, the instance field corresponding to the class field.

3. The method according to Claim 2, wherein said creating step (c) comprises creating, for each class field in the original class, at least one of an access function, a read access function and a write access function.

4. The method according to Claim 1, wherein the original class is transformed into one helper class.

5. The method according to Claim 1, wherein the original class is transformed into one modified class.

6. The method according to Claim 1, wherein transforming the class is applied to a byte code.

7. The method according to Claim 1, further comprising the step of loading the helper class and the modified class by use of the class loader when one of the processes is started.

8. The method according to Claim 1, wherein said converting substep (ii) further comprises introducing the original-class class-initialization method into the modified-original class and replacing the original-class class-initialization by an empty method.

9. The method according to Claim 1, wherein the helper-class class-initialization method creates a table.

10. The method according to Claim 1, further comprising the step of transforming an original interface, comprising at least one class field and/or an original-interface class-initialization method, into a modified interface and the helper class, comprising the substeps of:

(i) converting at least one said class field to an instance field and introducing it into the helper class; and

(ii) converting the original-interface class-initialization method to a helper-class instance-initialization method and introducing it into the helper class which comprises a helper-class class-initialization method,

wherein the helper class and the modified interface are loadable by the class loader.

11. Computer readable code stored on computer readable media for transforming a class in an object-oriented environment, comprising:

a first process for creating from an original class, which comprises a class field, an original-class class-initialization method, and a helper class, said first process comprising

first subprocessess for converting at least one said class field to an instance field and introducing the instance field into the helper class; and

second subprocesses for converting the original-class class-initialization method to a helper-class instance-initialization method and introducing it into the helper class which comprises a helper-class class-initialization method; and

a second process for creating for the class a corresponding modified class by converting the usage method to a modified-usage method, wherein each access to the class field is replaced by an invocation of an access function for fetching, for a process with an instance of the helper class, from the instance, the instance field corresponding to the class field,

the helper class and the modified class being loadable the class loader.

12. In a computing environment, a system for class transformation, said system comprising:

a class comprising at least one class field, an original-class class-initialization method, and a usage method accessing at least one of the class fields, said class residing in memory; and

a creator module for creating, out of said class, a helper class and a modified class,

wherein at least one said class fields is convertable to an instance field into said helper class, wherein said original-class class-initialization method is convertable to a helper-class instance-initialization method into said helper class which comprises a helper-class class-initialization method, and wherein in said usage method in said modified class each access to said class field is replaceable by an invocation of an access function for fetching the instance field corresponding to the class field for a process with an instance of said helper class, from said instance, and wherein said helper class and said modified class are loadable by a class loader.